Reference F: HUD Year 2000 Tools Overview

Table of Contents

List of Tables

Introduction

This reference document describes the software tools which HUD is using to support the Year 2000 conversion effort. Several groups at HUD are continually evaluating additional Year 2000 tools; this chapter will be updated as these tools become available.

Documentation

The tool products are documented in several ways:

➤ **Electronic Documentation**

- *On-line documentation* - Refer to the mainframe help system to locate online manuals for the HUD Year 2000 tools; the tools descriptions in this document contain information about on-line documentation availability.

- *Vendor Documentation* - Paper copies of vendor documentation are available through the Team 2000 project office. Efforts are underway to include electronic documentation on the Team 2000 HUDWeb site. If a desired document is not yet on the HUDweb, contact Team 2000 via our cc:Mail box at *Team_2000@hud.gov*.

➤ **HUD IT Technical Bulletins** have been produced for tool products which required a special introduction or announcement because of unique features. Technical Bulletins also tell you where to find additional technical information. When created, new bulletins are cc:Mailed to all IT post offices, to the Field Office ITD Directors and their staff members, and to the Year 2000 Office Technology Coordinators. They have also been posted on the HUDweb. Refer to **Reference Document D:** *Technical Bulletins* in the **Reference Library** for additional information.

Lists of Tools

The following two matrix tables will help you establish which tools are currently available for various functions. **Table F-1** is a matrix *of HUD Year 2000 tools by platform and tool name*. **Table F-2** is a matrix of *HUD Year 2000 tools and their languages and environments*.

Need Help?

For further information on (or assistance with) any of the tools mentioned in this document, please contact the **Year 2000 Assistance Service** at (202) 755-2000 or send an e-mail message to *Team_2000 @hud.gov*.

Table F-1:Year 2000 Tool Matrix by Platform and Product

	Vendor	Status	Phases Used			Function	Comments (Strengths/ Weaknesses /Recommendations)	Training Required
			Config. Manag.	Renov.	Testing			
HITACHI PLATFORM								
Abend-Aid	**Compuware**				●	➤ Analyzes the cause of abnormal application program terminations across a broad range of environments (for example, DB2, IMS, and IDMS databases).	➤ Isolates the problem. ➤ Performs a detailed analysis tailored to the specific abend. ➤ Gives programmers immediate online, menu-driven access. ➤ Supports DB2, IMS, IDMS environments.	Moderate
Comparex	**Serena**	HUD owned			●	➤ Increases productivity by automating data, text, and directory comparisons. ➤ Improves quality assurance by performing database comparisons in a single step. ➤ Expedites Year 2000 conversions by eliminating time-consuming manual verification of change.	**Automated Database Conversions:** ➤ Used to compare different files that are in different formats. Types of comparisons are Dynamic, Field-to-Field. **Year 2000 Comparison Utility:** ➤ Expedites impact analysis by identifying date fields within a program. ➤ Saves valuable programmer time by identifying missing source or load modules before modifications are made.	Moderate
File-Aid	**Compuware**			●		➤ Enables developers to quickly and easily fix problem data while maintaining security and data integrity.	➤ Improves productivity. ➤ Reduces production downtime. ➤ Provides environment flexibility. ➤ Supports wide range of environments..	Moderate
Parallel Development Manager	**Computer Associates**	HUD owned		●	●	➤ Helps developers perform coding activities concurrently without interfering with previous or simultaneous updates of application code.	➤ Concurrent work by a small number of programmers on modification to a single program. ➤ Concurrent work on multiple or overlapping releases by several teams of programmers.	Moderate
Platinum Repository	**Platinum**			●	●	➤ Identifies parameter fields in application code by using scanning and population tools.	➤ Maximizes use of existing resources by allowing older data and applications to work concurrently with newer technology and platforms.	Moderate
QAHiper-station	**Compuware**				●	➤ Streamlines the testing process and improves software quality. ➤ Performs comprehensive testing including unit, concurrence, integration, migration, and capacity and load tests.	➤ Automates the creation, execution, and analysis of test cases, allowing more test iterations to be performed within the established deadline. ➤ Advanced File-Aid/Data Ager; automates the process of incrementing, decrementing or specifying default values for day, month, year and century fields.	Moderate
Simulate 2000	**Prince Software**	HUD owned			●	➤ Product "sits" between application and system clock and provides a system date of your choice.	➤ Allows programmers to test applications using alternate dates—future or past. ➤ Helps locate application that contains date and time algorithms.	Moderate
Softaudit	**Isogon**	HUD owned	●			➤ Locates, inventories, and monitors applications. ➤ Determines which source and load modules are unused. ➤ Identifies which modules make up each load module and tells you where they are.	➤ Permits filtering by volume, source library, load library, module, application, user ID, jobname, date, time and job accounting data. ➤ Reports on every job or user using each application or each module. ➤ Software required is MVS/ESA (MVS/SP Version 3 Release 1 or later), or MVS/XA (MVS/S Version 2 Release 1.2 or later).	Moderate
System Vision 2000	**Platinum**	HUD owned		●		➤ Code scan and analysis tool. ➤ Provides estimates based upon number of dates in code. ➤ Identifies date related items in code.	➤ Mainframe-based product. ➤ Strong parsing engine.	Minimal
TransCentury Calendar Routines	**Platinum**	HUD owned		●	●	➤ Provides current date and time. ➤ Validates a specific date. ➤ Calculates age in Years, Months, and Days.	➤ Reduces time and expense of year 2000 conversions. ➤ Product compliant application that can easily be modified to reflect changing business rules.	Moderate

UNISYS PLATFORM									
I-QU 2000	**KMSystems**	HUD owned		●			➤ Code scan and analysis tool. ➤ Specifies unique formats for each date selected. ➤ Generates date field and record conversion code.	➤ Mainframe-based product. ➤ Can search , identify and convert any Unisys 2200 data format such as DMS-RDMS database, including LINC generated.	Minimal
Software Quality Assurance	**Arkdata AB**		●				➤ Easier way for programmers to maintain old systems. ➤ Supports and facilitates the work of developers and programmers. ➤ Designed to make your work easier by controlling the program changes, and transfer to production.	➤ You know that the source code f your program delivered by SQA is 100% the latest version. ➤ You have access to information on every change made to an application. ➤ Maintains clearly defined limit between the development and production environments. ➤ SQA can be moved to Clearpath. ➤ SQA over-administers everything. ➤ You have access to all previous generations of all source code. ➤ An information bank has been built up for Year 2000.	Moderate
TransCentury Calendar Routines	**Platinum**	HUD owned		●	●		➤ Provides current date and time. ➤ Validates a specific date. ➤ Calculates age in Years, Months, and Days.	➤ Reduces time and expense of Year 2000 conversions. ➤ Product compliant application that can easily be modified to reflect changing business rules.	Moderate
LAN PLATFORM									
PVCS Software Configuration Management	**INTERSOLV**			●	●		➤ Provides an infrastructure for software design, development and deployment.	➤ Can interface with various software through written interfaces by either INTERSOLV or partnered software companies. ➤ Supports LAN-LAN asynchronous replication of archives and unites the mainframe and client/server worlds.	Moderate
Visual SourceSafe	**Microsoft Corporation**			●	●		➤ Manages file and project relationships. ➤ Performs version control, keeping developers in sync. ➤ Supports parallel development.	➤ Easy to use and administer. ➤ Includes way to create your own custom source control applications and integrate them with existing development tools. ➤ Newer version, SourceSafe 5.0 has new features that will make software and website development much easier. ➤ Integration with a host of Microsoft tools and applications. ➤ Flexible tool, can manage any type of file.	Moderate

Table F-2: Year 2000 Tools Languages and Environment Matrix by Product and Platform

		Assembler	DB2	CICS	IDMS	IMS		PL/1	SQL	TSO	COBOL	FORTRAN	LINC	DMS	RDMS
Abend-Aid	Hitachi	●						●			●	●			
Comparex	Hitachi		●			●									
File-Aid	Hitachi					●	●								
Parallel Development Manager	Hitachi						●								
Platinum Repository	Hitachi						●								
QAHiperstation	Hitachi			●	●	●				●					
Simulate 2000	Hitachi										●				
Softaudit/2000	Hitachi						●								
SystemVision 2000	Hitachi	●		●		●		●	●		●				
TransCentury Calendar Routines	Hitachi & Unisys										●			●	●
I-QU 2000	Unisys										●	●	●	●	●
Software Quality Assurance	Unisys													●	●

Downdater/SSG

**(Unisys Platform Change
Detection and Merge Process)**

Product Overview   With the volume of changes needed for the Year 2000 effort, the ability to determine changes is crucial. This is especially true when auditing is done. The simplest precaution which can be taken is to copy the program files at the beginning of the change process. This copy can then be compared with the final version to identify all the changes. The Downdater (@DOWN) tool performs this comparison and produces a listing of all the changes

When parallel development occurs, the need arises to merge the changes from both development paths. The Symbolic Stream Generator (@SSG) allows the merging of two sets of corrections. SSG gives warnings when the two correction sets "collide" in the attempt to make different changes to the same code.

Note: The automated merge process detects only the errors which occur from a physical collision. It does not detect logical conflicts between changes. Testing is needed to identify any problems of that type.

Downdater also has the ability to compare the results of the two paths and identify elements added on one path and not the other.

Where Is It?   `PLIB*COMMON.FILE-COMP/SAMPLE` is a sample runstream.

Procedures   The process of determining and merging changes has the following steps:

1. Use Downdater to compare the baseline code with the results of one development path.
2. Use Downdater to compare the baseline code with the results of the other development path.
3. Use @SSG to merge the code and report on the "collisions".
4. Use Downdater to look for new elements on one path and not the other.
5. The results of steps 3 and 4 are analyzed and decisions are made on how to handle these cases.
6. Changes are applied to the baseline to produce the final change set.
7. The updated code is compiled and tested.

F-6

(This page has been left blank intentionally.)

File-AID

Product Overview   File-AID performs a variety of utility functions on files. It lists, searches, and compares files. File-AID supports the analysis and testing phases of the Year 2000 conversion. It helps improve the speed and quality of your testing, simplifying the steps involved in creating and validating test data by providing ISPF-like access to all file types (including VSAM, ISAM and sequential files of any record length) and by ensuring maximum flexibility in using the data once it is accessed.

File-AID works on the Hitachi platform; it does not run on the Unisys platform.

Product Functionality and Features   Some of the features of File-AID are:

➤ Global find capability allows programmers to quickly determine the magnitude of pending Year 2000 changes. It also searches source programs for all standard language syntax variations used to spell various types of dates.

➤ Selection feature finds records in all files and provides the number of instances that meet user-defined criteria. This helps find instances of data affected by a change.

➤ Uses logical JCL processing that allows it to apply complex changes to all members of JCL libraries, on-line or batch.

➤ Reformat facility can add and initialize new data fields and expand the length of existing data fields.

➤ Assists in system migration by converting data from one system to another.

➤ Compare facility lets programmers compare files to determine the differences.

➤ User-controlled audit file records all changes.

➤ Allows creation of customized test data.

➤ Monitors test DASD utilization.

How Do I Access / Invoke It?   From the ISPF main menu, enter (V)endor then (F)ile-AID. The File-AID main menu is then available.

F-8

(This page left blank intentionally.)

I-QU 2000

I-QU 2000, developed by KMSystems, Inc. as an add-on tool set for I-QU PLUS-1, enables database analysts to easily identify date fields in any DMS 2200 database. Based on the fields identified, the database analyst specifies the new date formats and I-QU 2000 generates the code necessary to convert the dates. I-QU 2000 automatically generates unload and reload reorganization runstreams, if necessary..

Product Overview

As an analysis tool, I-QU 2000 shows the effect changing records has on the database and provides a listing of affected areas and records. The database analysts can plan a phased conversion and minimize user downtime. I-QU 2000 also provides a list of programs that will be affected by the date conversions. The tool set can furnish date conversion information for other 2200 file structures, such as PCIOS and RDMS 2200.

Product Functionality

I-QU 2000 is a Year 2000 Date Conversion and Automatic Database Reorganization Generation Tool.

Where Is It?

There is an evaluation copy with Lockheed Martin Corporation. The Year 2000 Project Office purchased this tool on June 20, 1997.

How Do I Access / Invoke It?

Runstreams and call routines.

How Can I Use It?

I-QU 2000 provides the ability to:

➤ Search database schema or other definitions for any user-defined date field names.

➤ Specify unique formats for each date selected.

➤ Automatically generate date field and record conversion code.

➤ Search any Unisys 2200 data format such as DMS 2200, RDMS 2200(including LINC generated), PCIOS, and TIP/FCSS.

➤ Build unload and reload database reorganization programs for DMS 2200 data structures.

➤ Search program source libraries to identify programs that require modifications.

➤ Examine database information that indicates affected areas and records.

➤ Plan a controlled, effective conversion strategy.

Additionally, I-QU 2000 will give you the flexibility to reformat date fields to meet your standards.

Procedures

The Date Search Sign-On dialog contains parameters used to connect to the 2200 host.

Select Sign-On Script

This text box allows you to select a WinQ Sign-On Script file (.WQX). The sign-on script is required when connecting to the 2200 host. To develop a sign-on script, use the WinQ Script Manager program/icon located in the I QU 2000 program group. Complete on-line help for the WinQ Script

Manager is provided. A sample script (DATESRCH.WQS) is provided in the SCRIPTS directory of the I-QU 2000 installation directory (normally, IQ2K100).

Note: When using the WinQ Script Manager, it is necessary to specify an Application Ident. The Application Ident. for the search program is "DATESRCH". You may enter the complete drive, path and file specification directly into the box; select the script previously used to sign on; or use the Browse button to locate a new script.

Browse   Press this button to locate and select a WinQ Sign-On Script.

Optional Sign-On Parameters   This group box contains controls that may be used during the sign-on process. If the WinQ Sign-On Script supplies any of the values normally associated with host connection (Open Id., User Id., and so on.), then the corresponding parameters in this group are not required.

Open Id   In this text box, enter the string used on the $$OPEN to a DEMAND session at your site. If the open id. is embedded in the WinQ Sign-On Script (for example, SEND "$$OPEN DEMAND"), this parameter is ignored.

User Id   In this text box, enter your 2200 host user id.

Password   In this text box, enter your 2200 host password.

Enable WinQ Trace?   Check this box to enable a WinQ trace during sign-on. A separate window will appear tracing WinQ events that occur between WinQ's processing of the script and the host's responses.

Note: *Recommendation*: Test the WinQ Sign-On Script with the WinQ Script Manager (use the With Trace selection from the Test menu) before attempting to use the script with I-QU 2000

WinQ Trace to File   Check this box to send the WinQ trace to a PC file.

Trace File   In this text box, enter the drive, path and file specification for the WinQ trace file (for example, C:\IQ2K100\IQ2K.LOG).

Note: The drive and path must exist; however, if the file does not exist, it will be created.

OK   Press this button to begin the host sign-on process. Once connected and after a short pause while host processing (the search) completes, the Result of Date Search dialog will appear

Cancel   Press this button to cancel the sign-on and return to the I-QU 2000 - Date Search Criteria dialog.

Analysis Process   ► You start the analysis process by selecting an option that corresponds to the Unisys 2200 data model to be accessed from the Select Data Model group box.

➤ Enter Schema/Subschema Information for example:

- Schema Name - HUD1SCH
- Subschema Name - HUD1SUB
- Schema/Subschema File or TIP Code - UDS$$SRC*SCHABS
- $$ PROC File - If any

➤ Select Date Fields dialog contains lists of all records and fields included in the Data Item Index File created during the search phase of I-QU 2000.

➤ Enter Date Search Parameter Information. The schema/subschema/file specified above is searched for on the field names shown in the Data Search Parameter Group. You may add or remove field names in either the Include or Exclude data list.

➤ The search process creates a listing of the schema/schema/file fields that match the defined criteria. This result file becomes available for actual field manipulation (Add Fields, Change Fields, and Delete Fields).

F-12

(This page has been left blank intentionally.)

Parallel Development Manager

Parallel Development Manager (PDM), at tool from Computer Associates, is used to manage concurrent development and application maintenance activities. PDM is a powerful tool to help both developers and managers address the problems inherent in the steps of parallel development projects.

This tool runs under Hitachi platform, it does not run on the Unisys platform.

Product Overview

Parallel development is a term used to describe the concurrent development activities necessitated by the complexity of the application. There are three general kinds of parallel development:

1. **Simple parallel development** is the concurrent and independent development of applications by different programmers. Potential conflicts and overlaps need to be identified and resolved and the program modifications must be integrated before the program is placed in production.
2. **Complex parallel development** involves more than one programmer working individually on the same base program, with an eventual need to integrate the two sets of changes. Modified copies of a program must be integrated to produce a single copy. The potential for conflict tends to be higher, and the changes more difficult. Management intervention may be required to determine the best approach and the most appropriate resources to produce the final product.
3. **Vendor application updates** need to be maintained and updated on a regular basis. The user customizations to the base product must be identified and carried forward into the new release. Integrating concurrently developed software requires varying levels of involvement by programmers and managers. The integration process, analysis, to determine the scope of the integration project. Procedures need to be established to handle consolidation, conflicting problems and integrating the changes.

Product Functionality and Features

PDM eliminates conflicts between:

➤ Concurrent work by a small number of programmers on modification to a single program;

➤ Concurrent work on multiple or overlapping releases by several teams of programmers;

➤ Integrating customizations to a base release of vendor or internally developed software with a subsequent release of the software.

PDM produces a set of reports that managers can use for project planning. The reports are based on one or more Work-in-Process (WIP) files. PDM builds WIP files from a base file and one or several files derived from the base. The "build WIP" process analyzes the input files, identifies differences and conflicts, and creates statistics and reports. The build WIP process is the PDM mechanism for consolidating changes and identifying conflicts.

Once a WIP file has been edited to resolve conflicts, PDM can create an integrated output file by inserting and deleting lines based on the annotation in the WIP files, then write the integrated output file to a user specified location.

The *Parallel Development Manager* operates with standard PDS and sequential data sets, as well as Endevor elements.

PDM can be used in foreground or in batch. PDM processing involves three basic steps:

1. **Build the Work-in-Process (WIP) file**. PDM annotates the WIP file to clearly mark all insertions, deletions, and conflict areas. The build WIP process also produces reports and statistical data which allows a manager to estimate the time and resources to resolve the conflicts.
2. **Programmers edit the WIP files to resolve conflicts**. Editing tools provided by PDM allows a project team to try different edit scenarios before actually integrating the changes in the WIP file.
3. The final step in the PDM process is **to merge the WIP file into a merge output file**. The merge output file is a source file that can be stored in the appropriate source repository, input to a compiler, or added into Endevor.

Procedures

Overview:
- ➤ Logon to Hitachi mainframe by keying-in 'YTSO'
- ➤ Awaiting more information from CSG

Step by Step:
- ➤ **On-line Processing**
  - To use PDM, select it via the ISPF/PDF primary option menu by entering TSO PDM. The primary option menu is used to select each step of the PDM process.
- ➤ **Batch Processing**
  - To generate build WIP request statements for later job submission, select option 1 on the PDM Primary Option Menu.
  - Indicate B for batch processing in the processing mode. Provide the required Batch Specification information.
  - Press <Enter>; you will receive the same series of screens that appear when you process PDM in foreground mode.

PLATINUM Repository

Product Overview   PLATINUM Repository is a mainframe-based repository combined with an easy-to-use set of scanning and population tools. This DB2-based repository is accessible through a variety of user interfaces, including ISPF, OS/2 Presentation Manager and Microsoft Windows. It also offers an interface to PLATINUM's InfoPump, which provides bi-directional replication of data among heterogeneous databases.

Product Functionality and Features   PLATINUM Repository identifies parameter fields in application code. It also:

➤ Maximizes use of existing resources by allowing older data and applications to work concurrently with newer technology and platforms;

➤ Saves development time and cost by automating the transition from older, less flexible data sources to relational ones; no complex conversion routines need to be written and debugged;

➤ Simplifies data management and improves data access and ad hoc queries by converting and centralizing data into the DB2 relational structure; and

➤ Improves decision support by allowing administrators to create a comprehensive data warehouse, one that extracts, centralizes, and consolidates data from a variety of sources.

Where Is It?   PLATINUM Repository resides inside the DB2 subsystem.

How Do I Access / Invoke It?   You will need authorization to use this product. The Central Information Management's Central Database Administrator (CIM/CDBA) sets up the authorization and loads the year 2000 legacy information as a service. CIM also provides code analysis.

Procedures   ➤ Logon to Hitachi development mainframe by keying-in 'DTSO' at the HUD logo.

➤ Key-in the letter 'V' on the option line. This will put you at the vendor product selection menu.

➤ Key-in the letter 'Y' for year 2000. This will put you at year 2000 selection menu.

➤ Key-in SV for SystemVision.

(This page has been left blank intentionally.)

Simulate 2000

Simulate 2000 is a software product manufactured by Prince Software, Inc., which is designed to assist in the installation of Year 2000 conversion projects. *Simulate 2000* is an Hitachi-based component is used in the testing phase of the conversion effort. This tool cannot be used on the Unisys platform.

Product Overview

Simulate 2000 allows programmers to test applications using alternate dates—future or past. Alternate dates can be performed to batch jobs, TSO userids, started tasks and CICS terminals, transactions and database regions. *Simulate 2000* also has execution audit and CICS audit facilities to identify applications and transactions that access the system clock, and source code scan facilities to perform simple text scanning on source code.

Product Features

1. *Simulate 2000* helps locate applications that contains date and time algorithms through its Execution Audit Facility, CICS Audit Facility and Source Code Scan Utility.
2. Once identified, *Simulate 2000* can test applications that use the system clock by providing a modified system clock for the application.

Where Is It?

It is currently installed on the Hitachi System platforms (DSYS, YSYS).

How Do I Access / Invoke It?

After logging on the system, you can activate *Simulate 2000* in Batch three ways:

1. Add A //$altdate DD statement to the step you want to change the date.
2. Add A altdate= keyword on the exec statement of each step in a job.
3. Add A altdate= keyword on the job card to control the whole job.

Procedures

1. To Identify which application access the system clock through CICS Audit Facility and Source Code Scan Facility.
2. To test applications using alternate dates - future or past.

The product allows you to simulate testing of conversed mainframe applications. The test can be performed in batch, on-line (CICS, for example), and database regions.

Note: In order to completely understand the use of this product, it is highly recommended that you have a copy of the *Simulate 2000* User's Manual Version 2.2.

Batch

To use *Simulate 2000* in batch, you must code additional parameters on either the **JOBCARD** or the **EXEC** statement of the step you want to alter the **DATE/TIME**. If you want all steps in the job to use the alternate date then place the new parameters on the **JOB** statement. If you want to only alter the **DATE** on a specific step then place the new parameters on that step **EXEC** statement.

Batch Job Parameters Add the **ALTDATE=YYYY.DDD** and/or **ALTTIME=HH.MM.SS** to the **JOBCARD**

```
//JOBNAME  JOB (ACTNUM,ROOM), 'PGMRNAME' ,ALTDATE =
2001.031,
//                        ALTTIME=14.00.00
```

or for a specific step:

```
//STEPX  EXEC PGM=YOURPGM, ALTDATE = 2001.031,
ALTTIME = 14.00.00
//STEPLIB  DD
DSN=SYS.SIM2000.Y22.PTLRLOAD,DISP=SHR
```

CICS Simulate 2000 performs CICS testing with a modified date for the entire CICS test region.

As with batch job, *Simulate 2000* can establish a modified date for the region in several ways:

➤ With the *Simulate 2000* rules language,
➤ On a $ALTDATE DD statement in the CICS execution JCL, and
➤ On the Portal 2000 JCL extensions in the CICS execution JCL.

CICS transaction testing with alternate dates can be accomplished by having systems programming install *Simulate 2000* into the test CICS region. Once installed you can alternate DATE and/or TIME with the following CICS transactions.

```
Activate SIMULATE 2000 in the region PTLE ACTIVE -
to activate SIMULATE 2000
Setup the DATE and/or TIME and terminals to be
affected
     PTL Date Time TERMID
     will specify Alternate Date, Time, for TERMID
Where Date is in format YYYY.DDD (2001.032)  TERM27
Where TIME is in format HH:MM (11:35)
     TIME can also be specified as * to use the
     time from the system clock.
```

Database The DB2 module responsible for all date and time related functions uses the store clock (STCK) instruction to obtain the hardware clock. *Simulate 2000* provides a patch for the DB2 module. To use the patch, *Simulate 2000* was modified to support address space switch cross memory mode. This allows *Simulate 2000* to intercept the SQL request made by other regions, and access information from the caller region.

By specifying an alternate date from the CICS terminal, and the terminal issues a transaction that performs DB2 SQL requests, *Simulate 2000* now finds the alternate date used by the CICS terminal, and forces DB2 to use the date for the SQL request. By simply specifying an alternate date for a CICS terminal, any SQL calls to DB2 will reflect the date specified by that terminal.

(This page has been left blank intentionally.)

SystemVision 2000

SystemVision 2000, manufactured by the Platinum Corporation, automates pre-change analysis and sizes application resources and costs. It helps to build proactive plans and budgets, lowering their implementation costs and risks.

Product Overview

SystemVision 2000 provides a complete approach for analyzing, planning, and implementing changes necessary for Hitachi legacy mainframe systems to operate in the year 2000 and beyond.

SystemVision 2000's parsing technology first identifies all date occurrences and pinpoints the exact location of the affected source code. This is done on a system-wide basis, not just for individual programs. Parsing lets SystemVision determine how each occurrence is used: calculation, definition, screen or data transfer. This allows sophisticated reporting to assist management in selecting and implementing the most cost-effective change method.

SystemVision 2000 also includes a modifiable resource and cost table which is tailored to reflect your costs. This is used to produce different what-if budgets, reflecting alternate approaches to making the changes.

After approval of the budgets and selection of a change method, programmers are guided through the code step-by-step for insertion of ISPF EDIT macros to make the necessary changes.

Product Functionality and Features

Task-Sizing and Budgeting Features

SystemVision 2000 includes features that provide management with data to support its decisions when evaluating implementation options:

➤ **Estimate Reports:** SystemVision 2000's Estimate Report provides data to assist management in the budgeting and decision-making process. The report determines the total labor and CPU time to make each type of date change and test, then calculates the total cost for each application.

➤ **What-If Modeling***:* This feature enables management to evaluate the costs and time associated with different kinds of change methods: Field Expansion or Procedural Change.

➤ **System-Wide Analysis:** SystemVision 2000 examines each line of program source code, both data and procedural structures, and locates all year 2000-related items, including their definitions, data transfers, calculations, and comparisons.

➤ **Maintenance Difficulty Rating***:* The man-hours required to change and test a program are usually affected by its complexity. SystemVision 2000's unique parsing technology examines each program in the application, then extracts a variety of factors that are used to compute the program's maintenance difficulty rating. This rating is factored into the final change budget.

Code Changing Support Features

SystemVision 2000 takes the drudgery out of making code changes by simplifying and automating the entire process. The benefits provided by SystemVision 2000 include:

➤ **Complete Inventory of All Date Occurrences:** This enables users to know exactly what will have to be changed and where that line of code is located - for an individual application or for a complete system.

➤ **Fast and Accurate Pre-Change Analysis:** By automating the pre-change analysis, SystemVision 2000 reduces the most time-consuming aspect of the code change process. For example, an application of 300-500 programs is analyzed as a single unit of code and is usually completed in less than half a day from start to finish.

➤ **Increased Efficiency in the Code Change Process:** Senior analysts can review the inventory (impact analysis) reports and mark only those lines of code that need to be changed. The actual change can be made by lower-level analysts.

➤ **Code Change Support:** Changes to code are made using ISPF EDIT macros with the "point and click" method. SystemVision 2000 will "step" from one change location to the next. Programmers need only select an edit macro, press a PF key, and the change is instantly inserted into the source code.

Where Is It?

SystemVision 2000 is located on the Hitachi via (DSYS/YSYS) ISPF. It is not available on the Unisys platform.

How Do I Access / Invoke It?

After you logon, TYPE TSO SV from the ISPF menu, then press Enter. This will display the main menu for SystemVision.

TransCentury Date Routines

The Platinum Corporation's TransCentury Date Routines provide a full range of date conversion and date validation routines. They have been selected as the HUD standard for Year 2000 date handling. The routines are written in COBOL and will run on both Hitachi and Unisys platforms. The following is a general overview of the package and its capabilities.

Product Overview

TransCentury is Year 2000 compliant and full-featured. It handles a large variety of date functions and can replace most site date-handling code. A widely used product with more than 100 user sites, it is likely that a large percentage of the product's bugs have been identified. Site code is written, tested, and used only at that site. It can not be tested as extensively as a multi-customer package.

Product Functionality and Features

TransCentury is a set of COBOL routines which will run on any platform with a COBOL Compiler (including PCs and LANs).

The TransCentury Solution Set is composed of the TransCentury Calendar Routine and the Date Logic Generator (DLG). The Calendar Routines Package is a library of reusable, standard programming date modules that handles all of the most common business calendar date calculation. The DLG helps create and test "calls" to these modules.

Different sites and applications have different requirements and understandings of various terms and situations. TransCentury provides an extensive set of "tailoring" options" to let the user adjust it to suit their tasks. These options include the following:

1. **Processing Days Definition** will enable you to specify, for each day of the week, whether or not that is considered a processing (working) day.
2. **Holiday tables** can be used to specify dates that that you and your business partners consider "holidays."
3. **End-Point Definition** lets you specify whether either, neither, one, or both the start date and end date are to included in the calculation when computing the number of days between two dates.
4. **Century Default** lets you define a year that is to be considered the break point.
5. **Start of Fiscal Year and Fiscal Month Specification** offers a parameter that lets you specify the month when the fiscal year begin.
6. **Day of the week string** lets you define the numerical values that correspond to each day of the week.
7. **Date Masks** (for input and output) let you define the formats of the incoming and outgoing dates.

Where Is It?   TransCentury is currently installed on the Hitachi and Unisys mainframes.

Hitachi   On the Hitachi platforms, The TransCentury routines are in the following Partitioned Data Sets (PDS):

➤ **COPYBOOKS SYS2.TCDS.COPYBOOK**
➤ **LOADLIB   SYS2.TCDS.LOAD**

Unisys   On the Unisys Platform, PLIB*COMMON contains all the TransCentury materials:

➤ **COPY PROCS**
➤ **ASCII COBOL (ACOB) relocatables**
➤ **UCS COBOL (UCOB) object modules**

How Do I Access / Invoke It?   Programmer's "CALL" statement.   TransCentury Date Logic Generator assists the programmer by stepping them through the business rules required to generate a "CALL" statement through a series of screens.   Once the programmer is satisfied with the results, the Date Logic Generator will create a "CALL" statement in the original program source code so that, during execution, the base Calendar Routines products can be called with appropriately set parameters.

How Can I Use It?   TransCentury Calendar Routines (TRC) is designed to perform tasks, such as:

1. Providing the current date and time;
2. Validating a specific date;
3. Deciding if the specified date lies within a leap year;
4. Checking if the date is a holiday;
5. Determining if the specified date is a processing day;
6. Identifying the day of the week that corresponds to the specified date;
7. Converting a date from one format into another format (150 possible inputs and outputs);
8. Calculating the age in years, months and days;
9. Computing the number of days between two dates;
10. Adding or subtracting a number of days/weeks/months/years to or from a giving reference date.

YDATESUB
(Unisys Date Simulation)

This tool allows different users to concurrently run tests using different future dates. This simplifies the scheduling of testing because the system clock does not have to be reset.

Product Overview

This package lets ASCII COBOL programs retrieve a variable in the format (CCYYMMDD). A call to YDATESUB will return the simulated system date to a value which is set earlier in the run. The range of allowable test dates is 1980 to 2020.

The COBOL subroutine C$CURDATE and the ACCEPT from DATE YYYYMMDD command return the system date in an identical format. Therefore, when removing the simulation package, only the subroutine call needs to be changed.

Procedures

Executing YDATESUB

A program is executed to change the system date for any step which calls YDATESUB. This date value is stored in a temporary file. It lasts only for the duration of a @RUN. This works for batch and demand runs.

There are three steps involved in using this package:

1. Change your (ASCII) COBOL program to make the following call to retrieve the eight digit date:

```
01   THE-DATE.
03   CC   PIC 99.
03   YY   PIC 99.
03   MM   PIC 99.
03   DD   PIC 99.
..................
PROCEDURE DIVISION
CALL 'YDATESUB' using THE-DATE.
```

This replaces **ACCEPT** from **DATE** or calls to **C$CURDATE**.
The **YDATESUB** subroutine is in the file **PLIB*COMMON**.

2. Put the following in the ECL before the modified program is executed.

```
@XQT   PLIB*COMMON.MODYDATE
ccyymmdd
```

There is only one data card with the eight-character date. The test date must be in the range 1980-2020. The date is the date of the test program only. It does not limit the values of dates stored, retrieved and handled in the program.

If the date is invalid, **MODYDATE** will error with a descriptive message. The test date is stored in a temporary file named **PDATEFILE** for the subsequent executions.

This changes the date only for this run and only for those programs which use **YDATESUB**. Any program which uses **C$CURDATE** or the **COBOL ACCEPT** verb will get the actual system date.

Also, each run wanting a different date must have the execution of **MODYDATE** installed.

3. Execute the modified program and **THE-DATE** will be returned as the date provided in step two.

Removing YDATESUB   When testing is complete, the procedure to revert to the regular date is very straightforward.

1. Modify the COBOL program. Change the **'YDATESUB'** call to call **'C$CURDATE'** or to **ACCEPT date from DATE YYYYMMDD** to retrieve the 8 digit date from the system.
2. Recompile and remap.

The execution of **MODYDATE** is now irrelevant. The program will no longer read the file.

Note: *WARNING* - The reverse is not true. If you take out MODYDATE and leave the YDATESUB call in, the run will error.